

# AMENDMENTS TO THE CLAIMS

Please amend the claims of the present application as set forth in the listing of claims below. This listing of claims will replace all prior versions, and listings, of claims in the application.

Claims 1-52 were originally filed.

Claims 45 - 52 have been withdrawn.

Claims 7 - 14, 21, and 27 have been cancelled.

Accordingly, claims 1 - 6, 15 - 20, 22 - 26, and 28 - 44 are pending.

## LISTING OF CLAIMS

1. (Previously Presented) A method comprising:  
receiving a request to write data to a logical sector address of a flash memory medium;  
selecting a physical sector address from a list of free physical sector addresses;  
assigning the selected free physical sector address to the logical sector address forming a corresponding relationship between the addresses;  
storing the corresponding relationship between the addresses in a data structure; and  
writing the data into a physical sector of the flash memory medium at a location indicated by the selected free physical sector address.

2. **(Previously Presented)** The method as recited in Claim 1, wherein the data structure is stored on the flash memory medium.

3. **(Original)** The method as recited in Claim 1, wherein the data structure is contained in a random access memory device.

4. **(Original)** The method as recited in Claim 1, wherein the request is received from a file system.

5. **(Original)** The method as recited in Claim 1, further comprising storing the logical sector address in the physical sector of the flash memory medium along with the data.

6. **(Original)** The method as recited in Claim 1, further comprising:  
storing the logical sector address in the physical sector of the flash memory medium along with the data;

if the data structure is erased, then scanning the flash memory medium to locate the logical sector address stored with the data;

assigning the physical sector address containing the data to the logical sector address forming a reestablished corresponding relationship between the addresses; and

storing the reestablished corresponding relationship between the addresses  
in a new data structure.

7 - 14. (Canceled)

15. (Previously Presented) A computer-readable medium having computer-executable instructions for performing steps comprising:

receiving a request to write data to a logical sector address of a flash memory medium;

selecting a physical sector of the flash memory medium to store the data based on the ability of the physical sector to store the data without first being erased;

assigning a physical sector address of the selected physical sector to the logical sector address forming a corresponding relationship between the addresses;

storing the corresponding relationship between the addresses in a data structure;

writing the data into the physical sector; and

writing the logical sector address in the physical sector of the flash memory medium along with the data.

16. (Original) The method as recited in Claim 15, further comprising:

1 if the data structure is erased, then scanning the flash memory medium to  
2 locate the logical sector address stored with the data;

3 assigning the physical sector address containing the data to the logical  
4 sector address forming a reestablished corresponding relationship between the  
5 addresses; and

6 storing the reestablished corresponding relationship between the addresses  
7 in a new data structure.  
8

9  
10 17. (Original) The method as recited in Claim 15, wherein writing the  
11 logical sector address in the physical sector of the flash memory medium includes  
12 writing the logical sector address into a spare portion of the physical sector.  
13

14 18. (Original) The method as recited in Claim 15, wherein the data structure  
15 is contained in at least one memory device other than the flash memory medium.  
16

17  
18 19. (Original) The method as recited in Claim 15, wherein the data structure  
19 is contained in a random access memory device.  
20

21 20. (Original) The method as recited in Claim 15, wherein the request is  
22 received from a file system.  
23

24  
25 21. (Canceled)

22. (Original) A method comprising:

(a) receiving a request to write data to a logical sector address of a flash memory medium;

(b) assigning a physical sector address to the logical sector address forming a corresponding relationship between the addresses;

(c) storing the corresponding relationship between the addresses in a data structure;

(d) writing the data into a physical sector of the flash memory medium at a location indicated by the physical sector address;

(e) receiving a request to rewrite updated data to the logical sector address;

(f) assigning a new physical sector address to the logical sector address forming a corresponding relationship between the new physical sector address and the logical sector address;

(g) storing the corresponding relationship between the addresses from the aforementioned paragraph (f) in the data structure;

(h) writing the updated data into a physical sector of the flash memory medium at a location indicated by the new physical sector address; and

(i) marking the physical sector address from the aforementioned paragraph (b) as dirty.

1 23. (Original) The method as recited in Claim 22, wherein the data structure  
2 is contained in at least one memory device other than the flash memory medium.

3  
4 24. (Original) The method as recited in Claim 22, wherein the data structure  
5 is contained in a random access memory device.

6  
7 25. (Original) The method as recited in Claim 22, wherein the requests are  
8 received from a file system.

9  
10  
11 26. (Original) The method as recited in Claim 22, wherein the data structure  
12 is maintained by a flash abstraction logic of a flash memory driver.

13  
14 27. (Canceled)

15  
16  
17 28. (Original) A system, comprising:  
18 flash medium logic, configured to store data in a physical sector of a flash  
19 memory medium;  
20 a table, configured to map logical sector addresses received from a file  
21 system to physical sector addresses on the flash memory medium; and  
22 flash abstraction logic, configured to ascertain a next free physical sector on  
23 a flash memory medium and assign an address associated with the free physical  
24  
25

1 sector to a logical sector address associated with a write request received from the  
2 file system.

3  
4 29. (Original) The system as recited in Claim 28, wherein the flash  
5 abstraction logic is further configured to update the map of the logical sector  
6 addresses to the physical sector addresses, after assigning the address associated  
7 with the free physical sector to the logical sector address associated with the write  
8 request.

9  
10  
11 30. (Original) The system as recited in Claim 28, wherein the flash medium  
12 logic marks a physical sector address as dirty after contents associated with the  
13 physical sector address are rewritten to a new physical sector address.

14  
15 31. (Original) The system as recited in Claim 28, wherein the flash medium  
16 logic is further configured to store the logical sector address within a portion of the  
17 physical sector associated with the write request.

18  
19  
20 32. (Original) The system as recited in Claim 28, wherein the flash medium  
21 logic is further configured to:

22 store the logical sector address within a portion of the physical sector  
23 associated with the write request; and  
24  
25

1 scan the flash memory medium for the portion of the physical sector storing  
2 the logical sector address after initialization of the system.

3  
4 33. (Original) The system as recited in Claim 28, wherein the table is  
5 contained in a memory device other than the flash memory medium.

6  
7 34. (Original) The system as recited in Claim 28, wherein the table is  
8 contained in a random access memory device.

9  
10  
11 35. (Original) The system as recited in Claim 28, wherein the system is a  
12 flash driver system.

13  
14 36. (Previously Presented) A computer-readable medium for a flash driver,  
15 comprising computer-executable instructions that, when executed, direct the flash  
16 driver to:

17  
18 receive a request to write data to a logical sector address of a flash memory  
19 medium;

20 selecting a physical sector address from a list of free physical sector  
21 addresses;

22 assign the selected physical sector address to the logical sector address  
23 forming a corresponding relationship between the addresses;

24 store the corresponding relationship between the addresses in a table; and  
25



1 write the data into a physical sector of the flash memory medium at a  
2 location indicated by the physical sector address.

3  
4 37. (Original) A computer-readable medium for a flash driver, comprising  
5 computer-executable instructions that, when executed, direct the flash driver to:

6 receive a request to write data to a logical sector address of a flash memory  
7 medium;

8 assign a physical sector address to the logical sector address forming a  
9 corresponding relationship between the addresses;

10 store the corresponding relationship between the addresses in a table;

11 write the data into a physical sector of the flash memory medium at a  
12 location indicated by the physical sector address;

13 write the logical sector address in the physical sector of the flash memory  
14 medium along with the data;

15 if the table is erased, then scan the flash memory medium to locate the  
16 logical sector address stored with the data;

17 assign the physical sector address containing the data to the logical sector  
18 address forming a reestablished corresponding relationship between the addresses;

19 and

20 store the reestablished corresponding relationship between the addresses in  
21 a new table.  
22  
23  
24  
25

1 38. **(Previously Presented)** A system for tracking sectors in a flash memory  
2 medium, comprising:

3 means for receiving a request to retrieve data stored in the flash memory  
4 medium from a location indicated by a logical sector address;

5 means for locating a physical sector address corresponding to the specific  
6 logical sector address in a table; and

7 means for reading the data stored in the flash memory medium from the  
8 physical sector address located from the table.  
9

10  
11 39. **(Original)** The system as recited in Claim 38, further comprising:

12 means for storing the logical sector address with the data in the flash  
13 memory medium at a location indicated by the physical sector address;

14 means for reestablishing a portion of the table in the event of a power  
15 interruption, by scanning the physical sector address for the associated logical  
16 sector address; and

17 means for storing the logical sector address in the table at a location  
18 corresponding to the physical sector address.  
19

20  
21 40. **(Original)** The system as recited in Claim 38, wherein the request is  
22 received from a file system.  
23  
24  
25

1 42. (Original) The system as recited in Claim 38, wherein the table is  
2 maintained by a flash abstraction logic of a flash driver.

3  
4 43. (Original) The system as recited in Claim 38, wherein the means for  
5 reading of the data is performed by flash media logic of a flash driver.

6  
7 44. (Original) The system as recited in Claim 38, wherein the memory  
8 device is a type of random access memory.

9  
10  
11 45. (Withdrawn) A flash driver system, comprising:  
12 a free sector manager, configured to determine a next free physical sector  
13 address on the media and assign the address to a logical sector address of a write  
14 request received from a file system;  
15 a table, configured to store a map showing the assignment of the physical  
16 sector address to the logical sector address; and  
17 a flash medium logic, configured to write the data to the next free physical  
18 sector indicated by the free sector manager and store the logical sector address  
19 directly with the data on the flash memory medium.  
20

21  
22 46. (Withdrawn) The flash driver as recited in Claim 45, further  
23 comprising a flash abstraction logic, configured to scan the sectors of the flash  
24 memory medium for the logical sector address and note the physical sector address  
25

from which the logical sector address is stored and reestablish the map in the table in the event the table is erased.

47. (Withdrawn) The flash driver as recited in Claim 45, wherein the logical sector address is stored in a spare portion of the flash memory medium.

48. (Withdrawn) A flash driver system, comprising:  
a free sector manager, configured to determine a next free physical sector address available on the flash memory medium;  
a flash abstraction logic, configured to query the free sector manager for the next free physical sector address and link the physical sector address to a logical sector address received from a file system; and  
a table, configured to store the physical sector address to logical sector address linking performed by the flash abstraction logic.

49. (Withdrawn) The flash driver system as recited in Claim 48, wherein the flash abstraction logic is further configured to mark an existing physical sector as dirty, if a logical sector address received from the file system was previously linked to the logical sector address in the table.

50. (Withdrawn) The flash driver system as recited in Claim 48, wherein the flash abstraction logic is further configured to mark an existing

1 physical sector as dirty, if a logical sector address received from the file system  
2 was previously linked to the logical sector address in the table, but only after data  
3 associated with the logical sector address is successfully written to a new physical  
4 sector address on the flash memory media.

5  
6 51. (Withdrawn) The flash driver system as recited in Claim 48,  
7 wherein a portion of a physical sector is a status bit configured to indicate when a  
8 write operation is in progress.  
9

10  
11 52. (Withdrawn) The flash driver system as recited in Claim 48,  
12 wherein a portion of physical sector is a status bit configured to indicate when a  
13 write operation has been completed successfully.  
14  
15  
16  
17  
18  
19  
20  
21  
22  
23  
24  
25